

sonificADN

*Digital resource for the automatic generation of sound sequences
from genetic sequences*



Javier Forment

Musical notation for a sonification of a genetic sequence. The notation uses letters G, A, T, C to represent musical notes. Brackets are used to group notes across staves. The sequence of letters across the staves is:

System 1: G A T, A A A T, C T, G G T C, T T, A T T T, G C

System 2: G G A T, A A A C, G C T A, A C, T T A A, A C

System 3: G C T A, A C, G C T A, C C, G G T C, T T A A

System 4: G A T, A A A T, C T, A A A C, A T T T, A C

System 5: G G A T, A A A C, G C T A, T T A A, T T A A, A C

System 6: C T, G T C, A T T T, C C, G A T, A A A T

Teaching guide

sonificADN

*Digital resource for the automatic generation of sound sequences
from genetic sequences*



Daniel Carasso Fellowship project “Seed wars go digital: sustainability, big data and the social movement for open source seed systems” directed by Raquel Ajates (UNED).

Art Commission 1:

Marco Ranieri. Seeds in Motion

Art Commission 2:

Aina Socies, Lluís Vidaña, Nívola Uyà y Enric Socias. The dance of seeds

Art Commission 3:

Javier Forment. Sound Seeds

In collaboration with PLANEA Red de arte y escuela

Editorial design: No somos nada

English translation: Lucy Thorogood

Text edited by: Raquel Ajates



The contents of this publication may be distributed, remixed, and built upon for noncommercial purposes only, and so long as attribution is given to the creator and this licence is maintained. This guide is also available in Spanish.



Table of contents

SonificADN. Digital resource for the automatic generation of sound sequences from genetic sequences



1

Summary

2

Introduction

3

Background

4

Software

5

Installation guide

6

User guide

1 Summary

What does a gene sound like? How is a gene similar to a musical score? Can a gene create a song? This resource offers students and teachers the possibility of establishing and internalising both the concept of a genetic sequence and the concept of a piece of music by establishing a parallelism between a sequence of nucleotides (a gene) and a sequence of sounds (a song) using a computer programme. Given the multidisciplinary nature of the resource, which combines concepts of genetics, music and computer science, we propose its use through collaboration amongst teachers from the areas of Biology, Art and Technology.



Time: 2 hours

Venue: Classroom with Linux computers

sonificADN is an educational resource part of an artwork created within the art and science collaboration of the Daniel Carasso Fellowship research project “Seed wars go digital: sustainability, big data and the social movement for open source seed systems” directed by Raquel Ajates (UNED). The main objective of the project was to analyse the feasibility of alternative models of seed governance, exploring how the concept of the commons and digital open source movements can help protect seeds from an increasing loss of varieties and privatisation. It is essential to recognise the key role of seeds in maintaining the farmers’ rights, the sustainability of the food system, and the cultivated diversity in our fields and on our plates. The project included three artistic commissions that resulted in the creation of three artworks that expose with inspiration, creativity and great expertise, the social and environmental issues facing seeds and cultivated diversity, inviting us to reflect on their beauty and importance for human and planetary health, and calling for action to preserve them. The three artworks are: “Mobile Seed Bank” by Marco Ranieri, “The Dance of Seeds. The Germination of Knowledge” by the Associació de Varietats Locals, and “Sound Seeds” by Javier Forment. Each work has generated an open access interdisciplinary educational resource with multimedia material so that its impact goes beyond the exhibitions and reaches more audiences.

Introduction

Genes are the hereditary material of an organism, and play a key role in determining how it looks and functions. The DNA in each gene contains the specific instructions (like a recipe) for making a different protein, which are the building blocks of the cellular machinery of all types of living things. In the human body, each cell contains approximately 20,000 genes. Age and exposure to environmental factors, such as diet, exercise, drugs and chemicals, can turn genes on or off without changing the DNA sequence. The study of these non-DNA-altering changes is called epigenetics, a science that helps us bridge the gap between genes, the environment we live in, and our habits.

A gene is a long chain made up of many molecules linked together in sequence. You can also think of a gene as a long necklace made up of numerous beads. Just as a necklace might have pearls of, for example, four different colours, forming a multicoloured necklace, a gene is made up of four different molecules, known as nucleotides or bases: adenine (A), thymine (T), cytosine (C) and guanine (G). In this way, genes are differentiated from one another, not by their components, which are always those four bases, but by the order in which they are strung together, in the same way that two necklaces made up of exactly the same four-coloured beads are differentiated not by the beads they contain, but by the order in which they are inserted into the necklace.

For example, one gene could have the sequence AATCGAGTCATC... and another gene could have the sequence ATAATATATCGCGCGC.... Both are made up of adenines (A), thymines (T), guanines (G) and cytosines (C), but in a different order. They would be different genes with different functions in the organism.

Similarly, a song is made up of a sequence of sounds represented by notes. There are only 12 different notes in the chromatic scale, the Western musical scale we are used to: Do, Do#, Re, Re#, Mi, Fa, Fa#, Sol, Sol#, La, La# and Si (or, in the alternative Anglo-Saxon representation, C, C#, D, D#, E, F, F#, G, G#, A, A# and B). And yet, despite being only 12 notes, practically infinite melodies can be produced, which produce different emotions in the listener, thanks to chaining these notes in different sequences.

For example, one melody could be D-G-A-A-D-D (3 Ds, 2 As and 1 G) and another could be G-A-D-A-D-D (also Ds, 2 As and 1 G, but in a different order). These are different melodies that can give us different impressions when we listen to them.

To highlight this parallel between genetic sequences and sound sequences, the present device consists of a computer program that creates sound sequences from genetic sequences. Specifically, the program takes a genetic sequence provided by the user, and four or more sounds also provided by the user, and generates a sound sequence consisting of the provided sounds arranged in the same order in which the nucleotides are arranged in the selected genetic sequence.

As an example, if we provide the programme with the genetic sequence AATCGG and four piano sounds with the notes C, D, G and A, the programme will assign, for instance, the note C to base C, the note G to base G, the note A to base A and the note D to base T, and it will generate the piano melody formed by the sequence of the notes A-A-D-C-G-G. The guide on how to use the resource shows how to assign the different notes to each of the four bases A, T, C and G.

Or, if we provide the program with the genetic sequence TATCTG and four percussion sounds (e.g. bass drum, snare drum, cymbal and timpani), the program will assign, for example, the snare to base C, the bass drum to base T, the cymbal to base G and the timpani to base A and will generate the percussion rhythm formed by the sequence bass drum-timpani-base drum-snare drum-base drum-cymbal. The guide to the use of the resource indicates how to assign the different instruments to each of the four bases A, T, C and G.

In short, the programme turns the genetic sequence provided into a score that will be played by the instrument made up of the sounds we provide it with. This establishes another parallel between genetics and music, for just as a score guides musicians to produce one particular piece of music and not another, so genes guide cells to produce one particular organism and not others.

Specifically, the appeal consists of:

1

An open source software for the automatic generation of sound sequences from genetic sequences.

The program consists of a Python script (a small program consisting of a single text file with commands in the Python programming language) that takes a genetic sequence and a series of sounds and creates a sound sequence consisting of a succession of these sounds determined by the succession of nucleotides (A, T, C, G) in the genetic sequence.

2

An installation guide for this programme on a computer with an open source operating system (any version of Linux is suitable).

The installation guide details the process of installing the Python script on a Linux computer, as well as all the requirements necessary for its correct operation.

3

A guide to using the software, including how to obtain genetic sequences and how to obtain sounds for the software to combine them.

The user guide details (1) how to obtain genetic sequences from different organisms (e.g. seeds and animals), (2) how to obtain sounds, (3) where to place the files with the genetic sequences and the sounds you want to use, (4) how to associate the different sounds to the different nucleotides (A, T, C, G) of the genetic sequence, and (5) how to run the programme to obtain an .mp3 file with the corresponding sound sequence.

Background

The resulting sound pieces are part of the current of musical creation known as “generative music”, “procedural music”, “algorithmic music”, “aleatoric music”, “serialism”, or similar, with representatives such as John Cage, Arvo Pärt, Karlheinz Stockhausen, Edgard Varèse, Iannis Xenakis, Pierre Boulez, Charles Ives, Henry Cowell, Earle Brown, Cornelius Cardew, Elliot Carter, Alvin Lucier, Terry Riley, Steve Reich, Joseph Schillinger or Lejaren Hiller. In this type of music, the piece is generated automatically from a series of initial musical elements and a series of pre-established rules and formal processes. In this form of creation, the composer is more interested in defining these conditions and letting the piece be generated from them without his or her intervention, than in creating a piece of music from beginning to end according to aesthetic criteria. The aim is therefore the process rather than the result, so that the resulting pieces may deviate to a greater or lesser degree from the usual musical aesthetics.

Examples:

- **Music of Changes, by John Cage**, based on permutations drawn from the “Book of Mutations” (I Ching):
<https://sulponticello.com/iii-epoca/john-cage-y-el-i-ching>
https://www.youtube.com/watch?v=B_8-B2rNw7s
- **Le Marteau Sans Maître, by Pierre Boulez**, an outstanding representative of the musical style known as serialism:
<https://www.youtube.com/watch?v=4uxwxOdb8P0>
- **Studie I y Studie II, by Karlheinz Stockhausen**, both with a compositional approach based on processes of serialisation of physical parameters and on proportions derived from mathematical operations, but with different sonic results:
<https://www.youtube.com/watch?v=H4QaMwpVXVM>
<https://www.youtube.com/watch?v=bwj6ZptPnDo>
- **Metastaseis, by Iannis Xenakis**, composed on the basis of Le Corbusier’s architectural geometric progressions or Fibonacci’s mathematical progression:
<https://es.laphil.com/musicdb/pieces/2339/metastaseis>
<https://www.youtube.com/watch?v=SZazYFchLRI>
- **Tabula Rasa, by Arvo Pärt**, a leading exponent of what is known as generative music:
<https://www.youtube.com/watch?v=8HON4AswPVk>

Software

The program can be downloaded in a .zip file from the URL <https://redplanea.org/recursos/recursos/sonificadn/>. This .zip file contains both the Python script (the file 'sonificADN.py') and its configuration file (the file 'config.txt'), an example genetic sequence (the file 'seq.fasta' in the directory 'seqs') and a series of sample sounds to use in the sonification (the files in the directories '0.perc', '1.inst' and '2.voz' in the directory 'sounds'). With these sample files of genetic sequence and percussion sounds, one instrument and some voices, the programme can be used directly to sonify the genetic sequence using these sounds. Below is a description of how to obtain other genetic sequences and other sounds to experiment with other possible sonifications.

PLANEA

CENTRO DE RECURSOS

¿Qué es el centro de recursos? Requisitos Colecciones Participo Revistas Comité

BUSCAR RECURSOS

Inicio / Recursos / SonificADN

SonificADN

Recurso informático para la generación automática de secuencias sonoras a partir de secuencias genéticas

Recurso en torno a la guitarra flamenca preparada, un concepto vinculado al piano de John Cage y a otros artistas y prácticas experimentales del siglo XX y XXI. A lo largo del recurso se explica cómo prestar atención a las modificaciones que sufre el sonido de las cuerdas utilizando algo tan simple como la masilla que nos sirve para pegar papeles, el Blu Tack, o un puente de bambú que cambia el timbre de las cuerdas, palillos que nos permiten percudir el instrumento o un ventilador de mano modificado para conseguir trémolos infinitos. No hay necesidad de que el alumnado esté previamente vinculado con la guitarra y puede usarse por pequeños grupos. A través de esta experiencia vamos a descubrir y a preguntarnos cuestiones como ¿Qué es la música? ¿Cómo "se hace" la música? ¿Qué idea tenemos sobre qué es "la melodía"? ¿Puede entenderse la vibración de un golpe como música? ¿Qué necesito para hacer música? ¿Qué otras prótesis musicales conocemos? El acercamiento experimental al instrumento conecta con una tradición y un patrimonio que nos hace recordar que en cuestiones de aprendizaje no solo hay un camino posible.

Este recurso se puede distribuir, copiar y remezclar citando la fuente y manteniendo esta licencia.

COLECCIÓN

Destripando la tecnología para crear

PLANEA

¿Qué necesitas?
Un ordenador y conexión a internet

Nivel formativo
Bachillerato, ESO,

Materia
Educación musical, Lengua y literatura, Matemáticas, Tecnología, Transversal

Formato
Documento

Idiomas
Español

Fecha

Installation guide

5.1.

Installation is required on a computer running the open source Linux operating system, preferably the Ubuntu Linux distribution. Alternatively, a Windows or Mac computer with a Linux virtual machine on it, e.g. VirtualBox (<https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-using-virtualbox>) could be used. In principle the programme could also be installed and used on a Windows or Mac computer, but this resource has been developed on Linux and there is no guarantee that it will work properly on Windows or Mac. The instructions in this guide are therefore aimed at installation and use on Linux.

5.2.

Download and unzip the .zip downloaded in point 5.1. This will generate a directory called 'sonificADN' with the Python program and its configuration file, as well as two subdirectories ('seqs' and 'sounds') with examples of a genetic sequence and different percussion, guitar and voice sounds, respectively, to use directly without the need to download new sequences and different sounds.

5.3.

Install the Python library 'musicpy' (<https://musicpy.readthedocs.io/en/latest/>) following the steps indicated in the 'Installation' section of <https://musicpy.readthedocs.io/en/latest/Introduction/>

5.4.

Install the 'daw' module of 'musicpy' following the steps indicated in the 'Preparation before importing' section of <https://musicpy.readthedocs.io/en/latest/Musicpy%20daw%20module/>

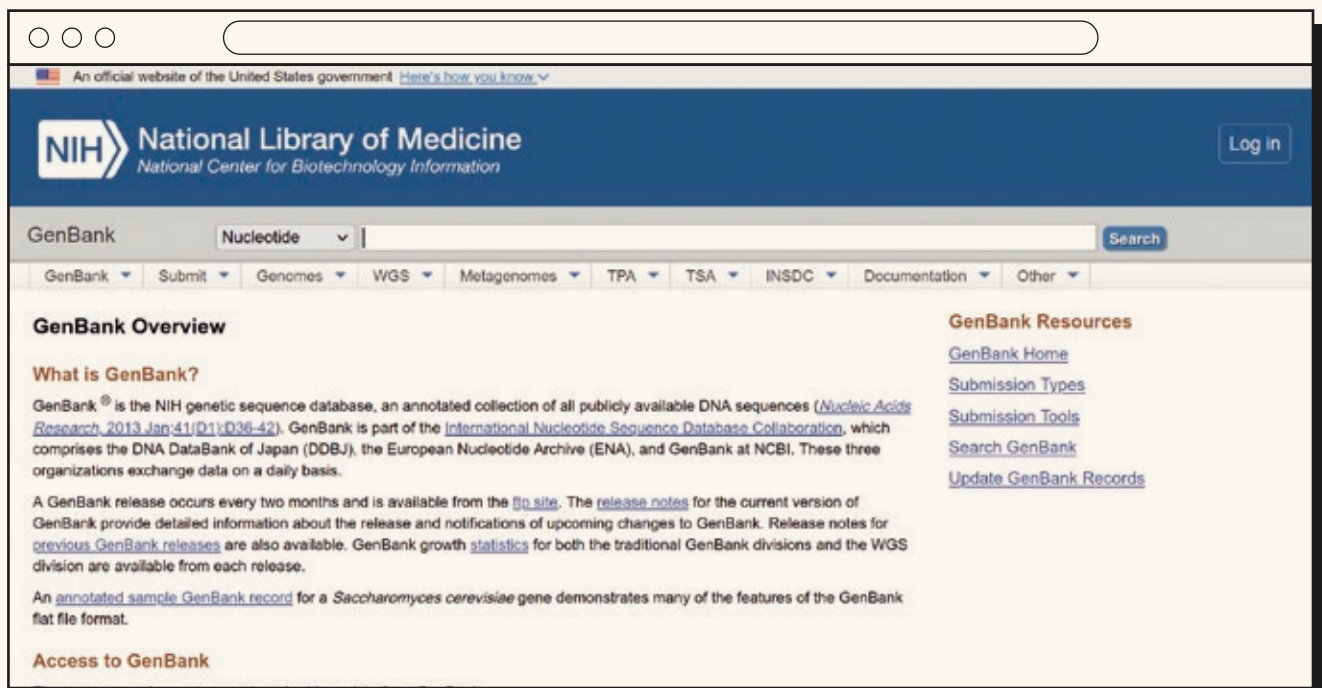
User Guide

6.1. Obtaining genetic sequences from different organisms

There are numerous public databases on the internet from which a multitude of genetic sequences from different organisms can be freely downloaded. The main one is the international repository of genetic sequences GenBank (<https://www.ncbi.nlm.nih.gov/genbank>). To download genetic sequences from GenBank and use them with sonificADN, the following steps can be followed:

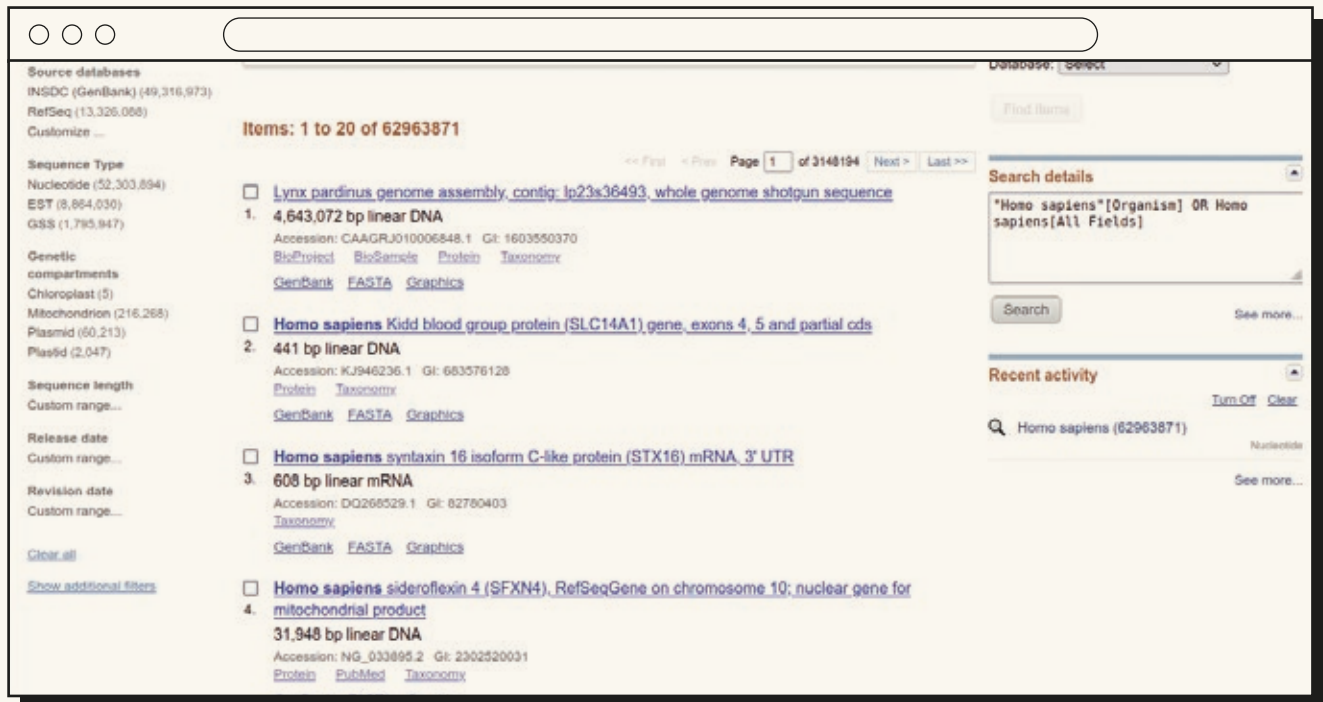


Click on: <https://www.ncbi.nlm.nih.gov/genbank>



- a) In the 'Search' box above, select 'Nucleotide' (this is selected by default) and enter a search term (e.g. 'Homo sapiens' to search for human sequences).

- b) After clicking on the “Search” button, a list of database items containing the search term will be displayed.



- c) Many of the results contain very long sequences that would take the program a long time to sonify, so it is recommended to filter the results by sequence size using the filter tool on the left of the page: “Sequence length - Custom range...”, to select sequences from a more manageable size range. A 200 nucleotide sequence (such as the one provided as an example along with the program) takes the program less than a minute to sonify on a standard laptop and produces a 53 second sound sequence using a 150bpm rhythm (as indicated in the configuration file provided along with the program). Testing with different sequence sizes and different bpm is recommended.



d) After filtering by sequence size, a list of items within that size range will be displayed.

The screenshot shows the NIH Nucleotide search interface. The search criteria are set to 'Nucleotide' and 'Homo sapiens'. The results are filtered by sequence length from 100 to 1000. The first two results are:

- Homo sapiens Kidd blood group protein (SLC14A1) gene, exons 4, 5 and partial cds**
Accession: KJ946236.1 | GI: 683576128
Links: Protein, Taxonomy, GenBank, FASTA, Graphics
- Homo sapiens syntaxin 16 isoform C-like protein (STX16) mRNA, 3' UTR**
Accession: DQ268529.1 | GI: 82760403
Links: Taxonomy, GenBank, FASTA, Graphics

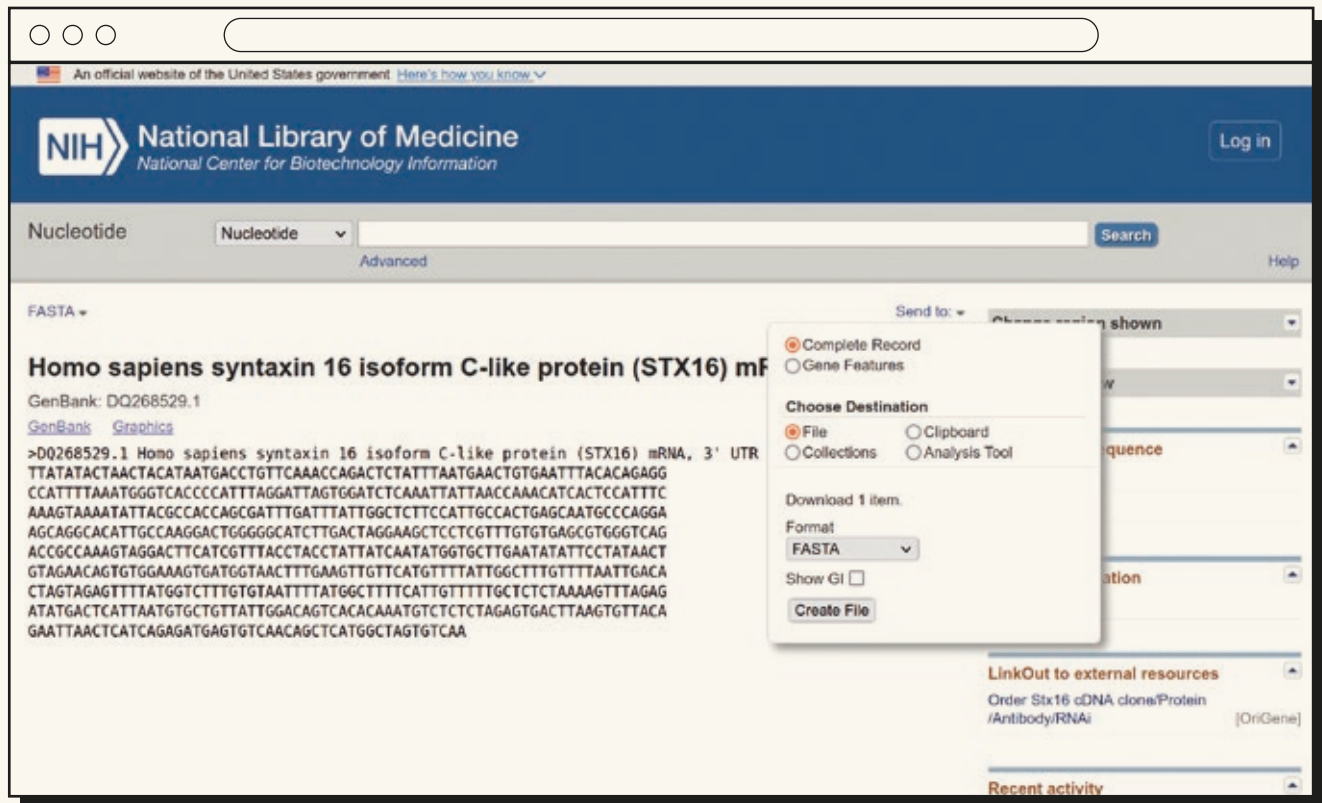
e) To download the sequence of one of them, use the link “FASTA” below it. After clicking on “FASTA”, the selected gene sequence will appear.

The screenshot shows the FASTA view of the selected sequence. The sequence is displayed in a text box, and the page includes links for 'GenBank', 'Graphics', 'Analyze this sequence', 'Run BLAST', and 'Pick Primers'.

Homo sapiens syntaxin 16 isoform C-like protein (STX16) mRNA, 3' UTR
GenBank: DQ268529.1
Links: GenBank, Graphics

>DQ268529.1 Homo sapiens syntaxin 16 isoform C-like protein (STX16) mRNA, 3' UTR
TTATATACTAACTACATAATGACCTGTTCAAACAGACTCTATTTAATGAACGTGAATTTACACAGAGG
CCATTTTAAATGGGTACCCCATTTAGGATTAGTGATCTCAAATTAACCAACATCACTCCATTTT
AAAGTAAATATTACGCCACGAGGATTTGATTTATTTGGCTCTTCATTGCCACTGAGCAATGCCAGGA
AGCAGGCACATTGCCAAGGACTGGGGCATCTTGACTAGGAAGCTCTCGTTTGTGTGAGCGTGGGTGAG
ACCGCCAAAGTAGGACTTCATCGTTTACCTACCTATTATCAATATGGTCTTGAATATATTCCTAACT
GTAGAACAGTGTGAAAGTGATGTTAACTTTGAAGTTGTTTATTTGCTTTGTTTAAATTGACA
CTAGTAGAGTTTATGGTCTTTGTGTAATTTATGGCTTTTCATTGTTTGTGCTCTCTAAAGTTTAGAG
ATATGACTCATTATGTGCTGTTATTGGACAGTCACACAAATGCTCTCTAGAGTGACTTAAGTGTACA
GAATTAACATCATCAGAGATGAGTGTCACAGCTCATGGCTAGTGTCAA

- f) To download it to your computer, use the “Send to” link at the top right, select “Complete Record” and “Choose Destination → File”. Then select “Format → FASTA” and click on “Create File”.



- g) Depending on your we have configured the browser, we will be asked to select a name for the file with the sequence, as well as a directory on our computer to download it to, or the browser will directly download the sequence in the directory assigned for downloads.

On the other hand, sonificADN could also sonify any genetic sequence you may have in your computer, whether it is real (downloaded from a public database as indicated above) or invented. Another option, then, is to create a genetic sequence by writing an invented sequence of letters A, T, G and C in a text file. The only requirement is to use a simple text editor for this (on Ubuntu you can use ‘gedit’, or ‘nano’) to save the sequence in plain text format and not in any other format that includes information beyond simply text characters (such as colour, font size or type, or margin size, italics, bold and other layout information) and not to include characters other than A, T, C and G.

6.2. Obtaining sounds for the sonification

In theory, any set of .wav or .mp3 files could be used to sonify a sound sequence. It is best, however, to use .wav or .mp3 files that contain a single, short sound each, to avoid the excessive overlap that would occur between long sounds when chaining them together in the sound sequence when the duration of the sound is longer than the interval between them.

There are different sound databases from which to download sound files with these characteristics. One of these databases worth highlighting is Freesound (<https://freesound.org>), as it allows you to upload and download audio files under a Creative Commons licence. To download sound files from Freesound and use them with sonificADN you can follow the following steps:

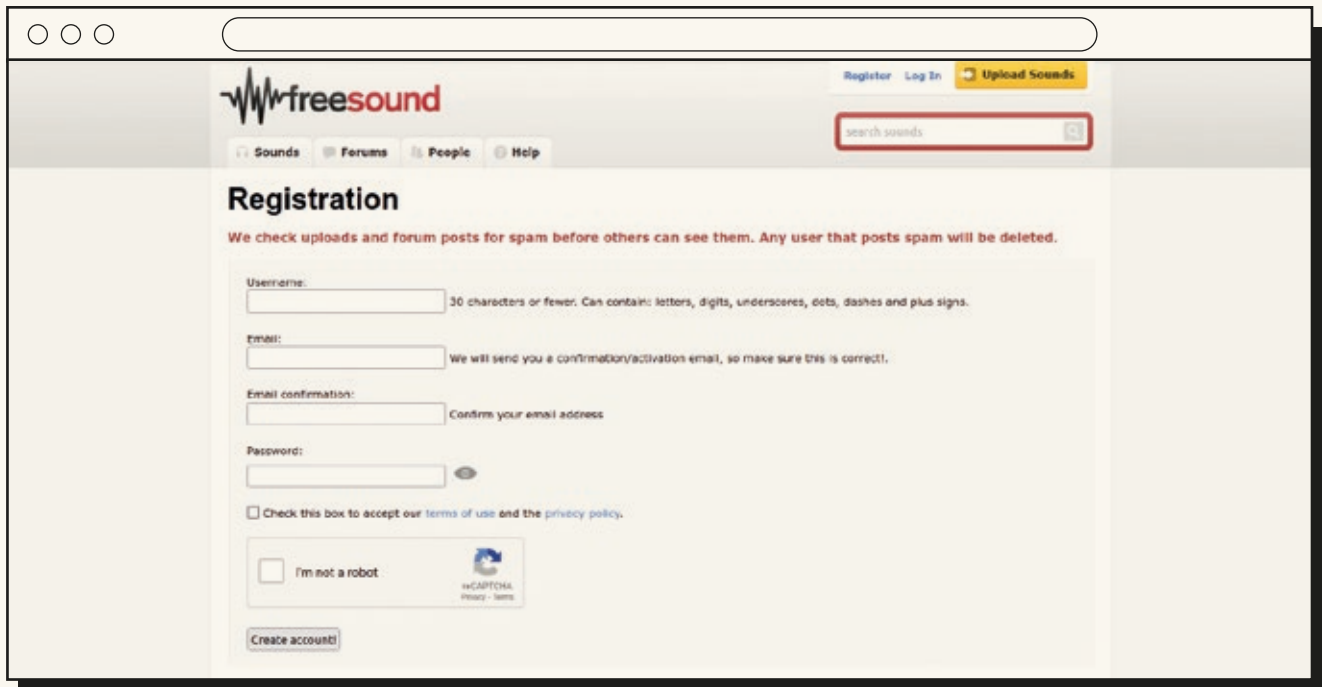


Click on: <https://freesound.org>

The screenshot shows the Freesound website interface. At the top, there's a navigation bar with the Freesound logo, a search bar, and links for Register, Log In, and Upload Sounds. Below the navigation bar, there are tabs for Sounds, Forums, People, and Help. The main content area is divided into several sections:

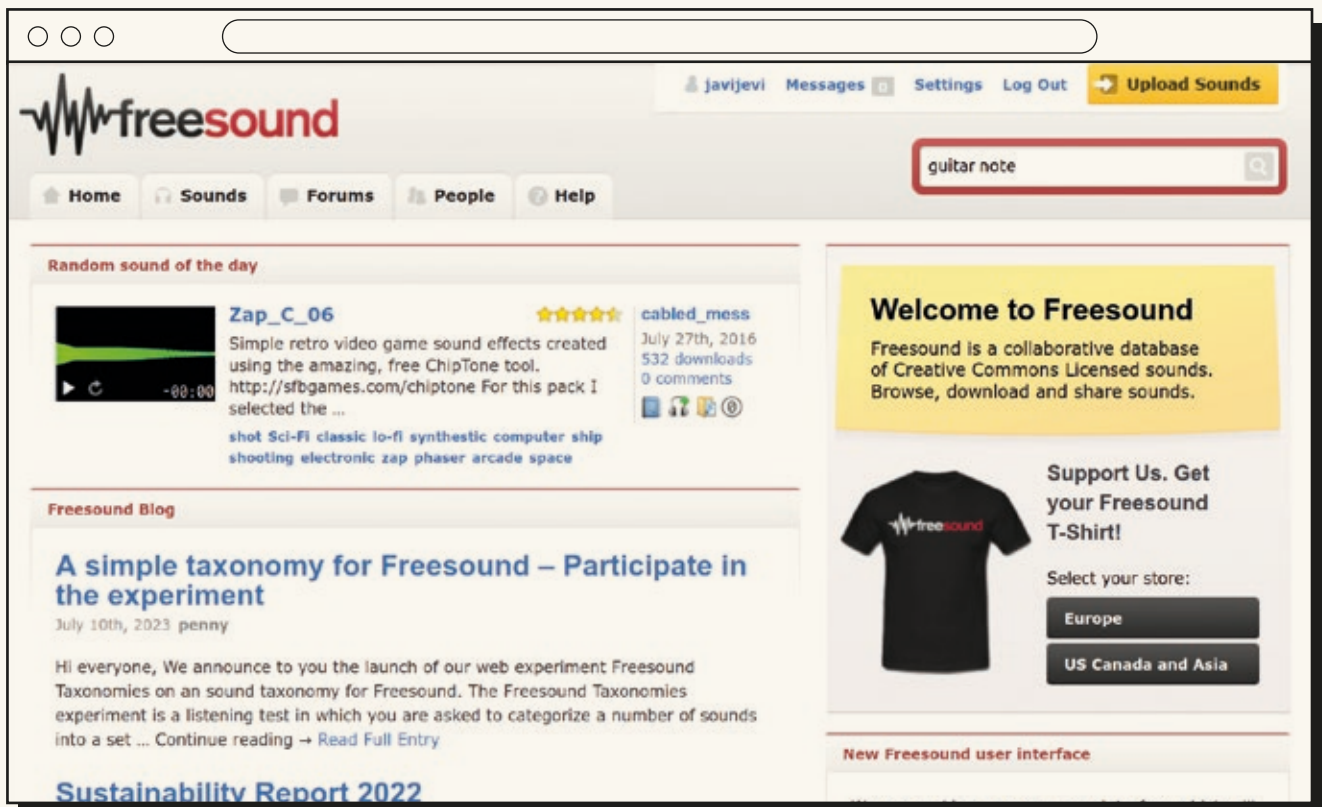
- Random sound of the day:** Features a sound titled "Zap_C_06" by "cabled_mess". It includes a waveform visualization, a description of the sound (Simple retro video game sound effects), and a link to the creator's profile.
- Freesound Blog:** Contains two blog entries:
 - A simple taxonomy for Freesound – Participate in the experiment:** A post by penny dated July 10th, 2023, announcing a web experiment.
 - Sustainability Report 2022:** A post by frederic.font dated April 28th, 2023, providing an update on the project's sustainability.
- Welcome to Freesound:** A yellow banner stating that Freesound is a collaborative database of Creative Commons Licensed sounds.
- Support Us. Get your Freesound T-Shirt!** A section promoting a t-shirt with the Freesound logo, with buttons to select the store (Europe or US Canada and Asia).
- New Freesound user interface:** A section announcing a new user interface and providing a link to "Try the new UI".

- a) In order to download audio files from Freesound you need to be registered and logged in. If you are not registered with Freesound you can do so using the “Register” link at the top right. Registration is free and only requires choosing a username, providing a valid email address to receive a registration confirmation link and choosing a password to log in.



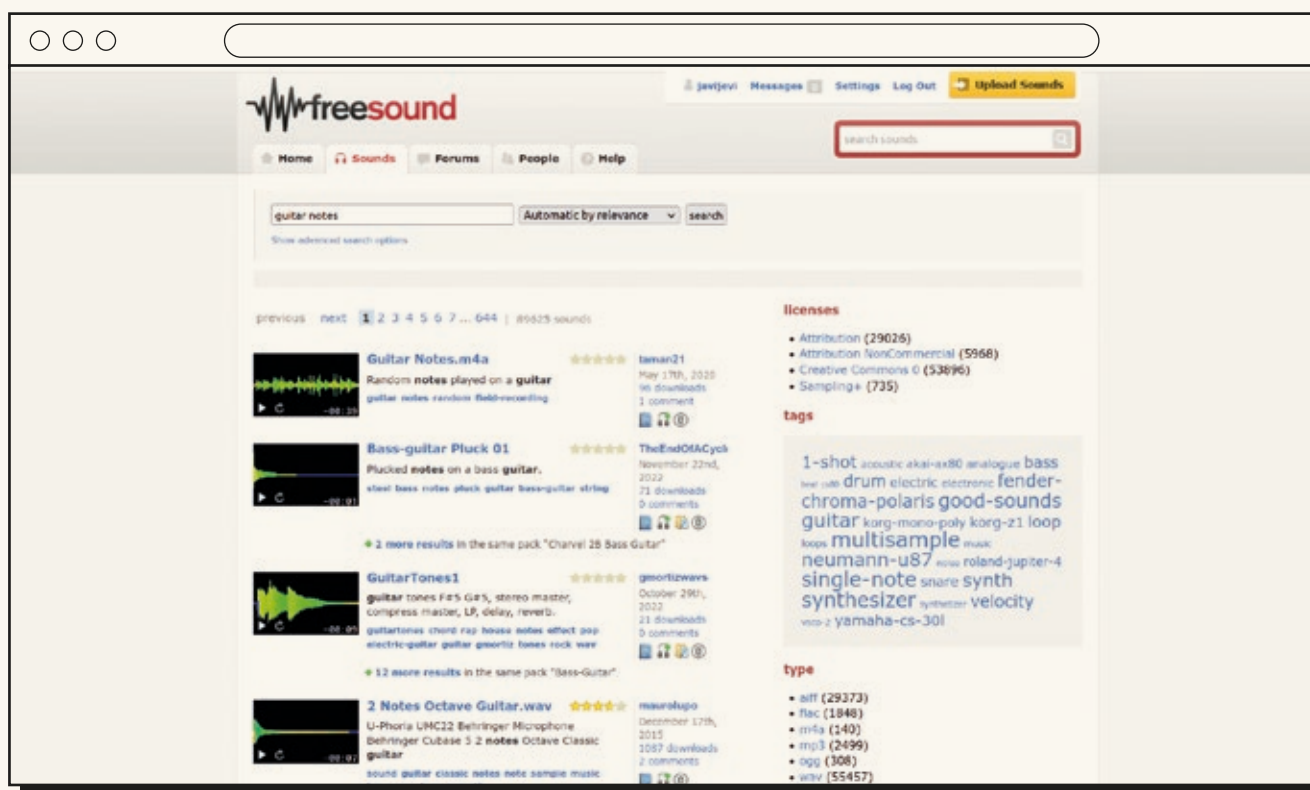
The screenshot shows the Freesound website's registration page. At the top, there is a navigation bar with links for "Register", "Log In", and "Upload Sounds". Below this is a search bar labeled "search sounds". The main heading is "Registration". A note states: "We check uploads and forum posts for spam before others can see them. Any user that posts spam will be deleted." The registration form includes fields for "Username:" (with a note: "30 characters or fewer. Can contain: letters, digits, underscores, dots, dashes and plus signs."), "Email:", "Email confirmation:", and "Password:". There is a checkbox for "Check this box to accept our terms of use and the privacy policy." and a CAPTCHA section with the text "I'm not a robot" and a "Create account" button.

- b) After logging in, to search for sounds, enter a search term (e.g. 'guitar notes' to search for single guitar notes) in the 'search sounds' box at the top right.

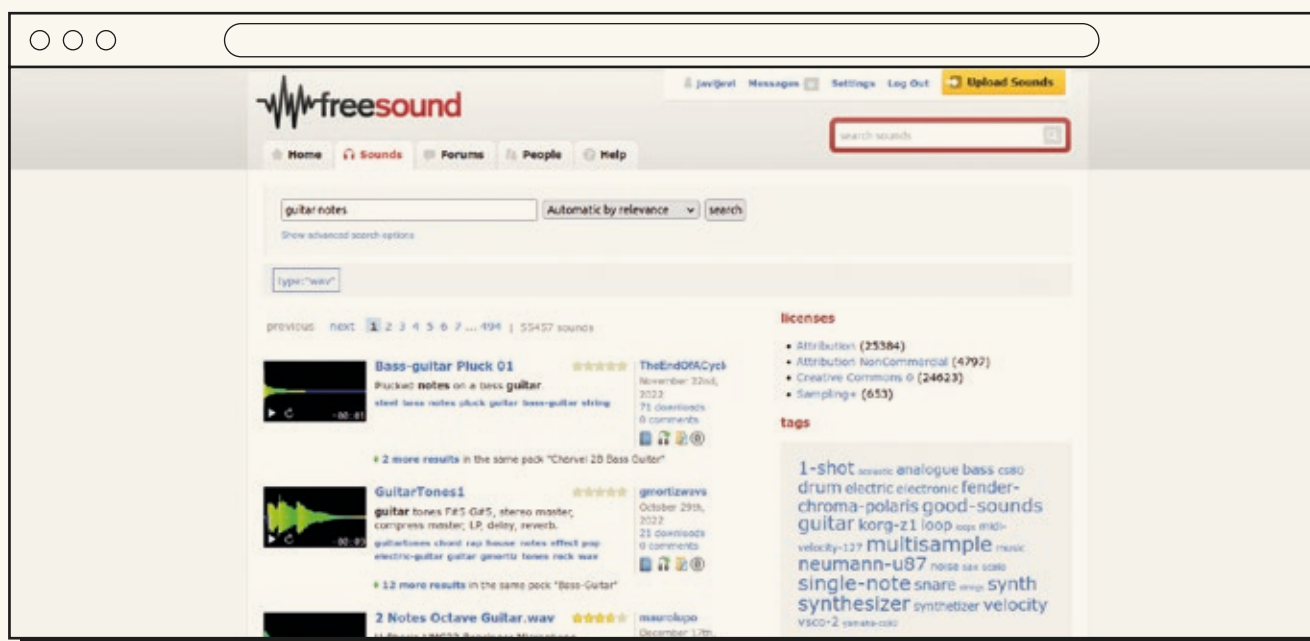


The screenshot shows the Freesound homepage after logging in. The navigation bar includes links for "javijevi", "Messages", "Settings", "Log Out", and "Upload Sounds". The search bar now contains the text "guitar note". The main content area features a "Random sound of the day" section with a sound player for "Zap_C_06" by "cabled_mess". Below this is a "Freesound Blog" section with a post titled "A simple taxonomy for Freesound – Participate in the experiment" by "penny". On the right side, there is a "Welcome to Freesound" banner and a "Support Us. Get your Freesound T-Shirt!" section with buttons for "Europe" and "US Canada and Asia". At the bottom, there is a "New Freesound user interface" section.

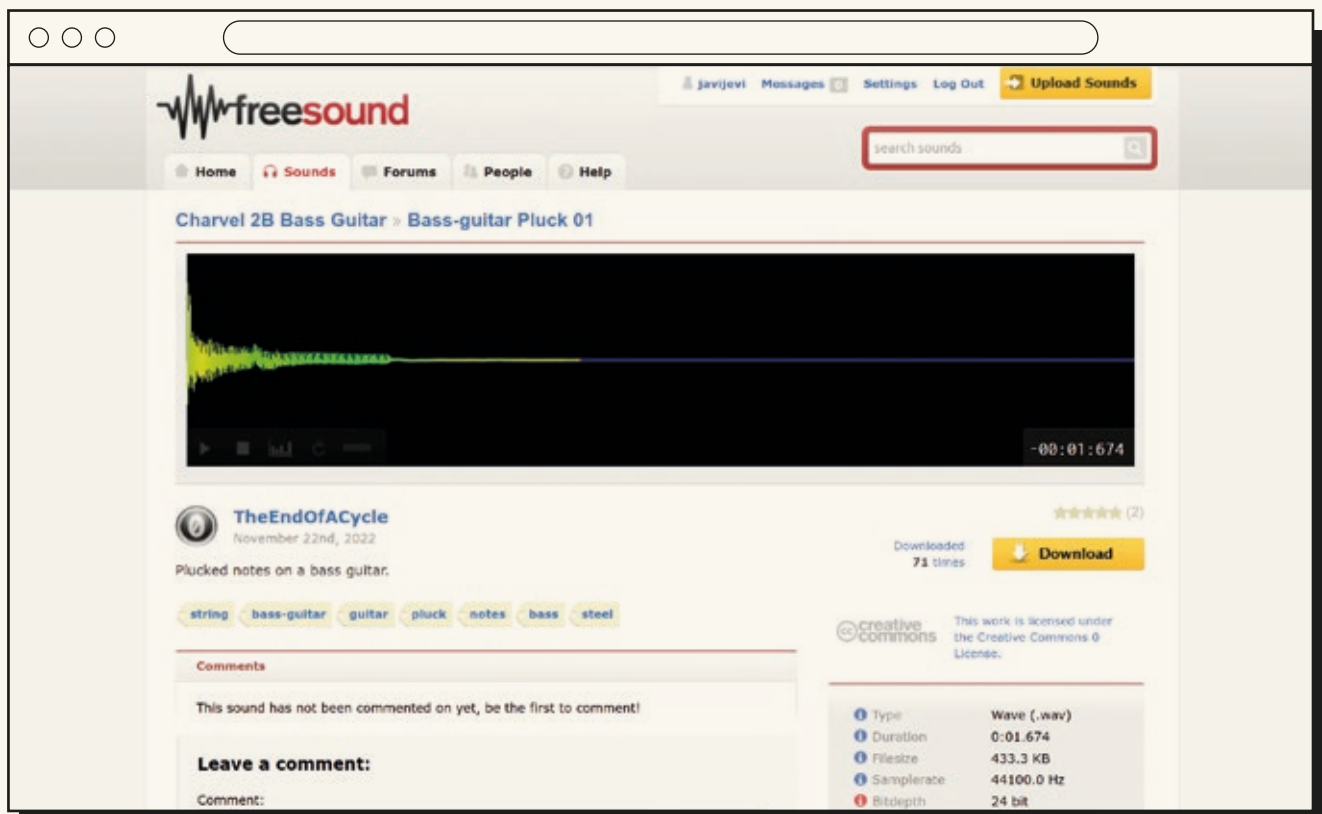
- c) After clicking on the magnifying glass icon to start the search, a list of items in the database containing the search term will be displayed.



- d) These items contain sounds in different formats (.wav, .mp3, .flac, .ogg, etc.), so it is recommended to use the filtering tool on the right to select only .wav format sounds (although .mp3 and .ogg format sounds should be usable as well).
- e) After filtering by audio format, a list of items in that format will appear. In the image next to the title of each item, a “Play” icon appears, which can be used to listen to the sound of the item and decide which one you would like to download.



- f) To download the file of the selected sound, click on the title of the audio you want to download and then click on the “Download” button that appears on the right.



- g) Depending on how browser settings, you will be asked to select a name for the file with the sequence, as well as a directory on our computer to download it to, or the browser will directly download the sequence in the directory assigned for downloads.

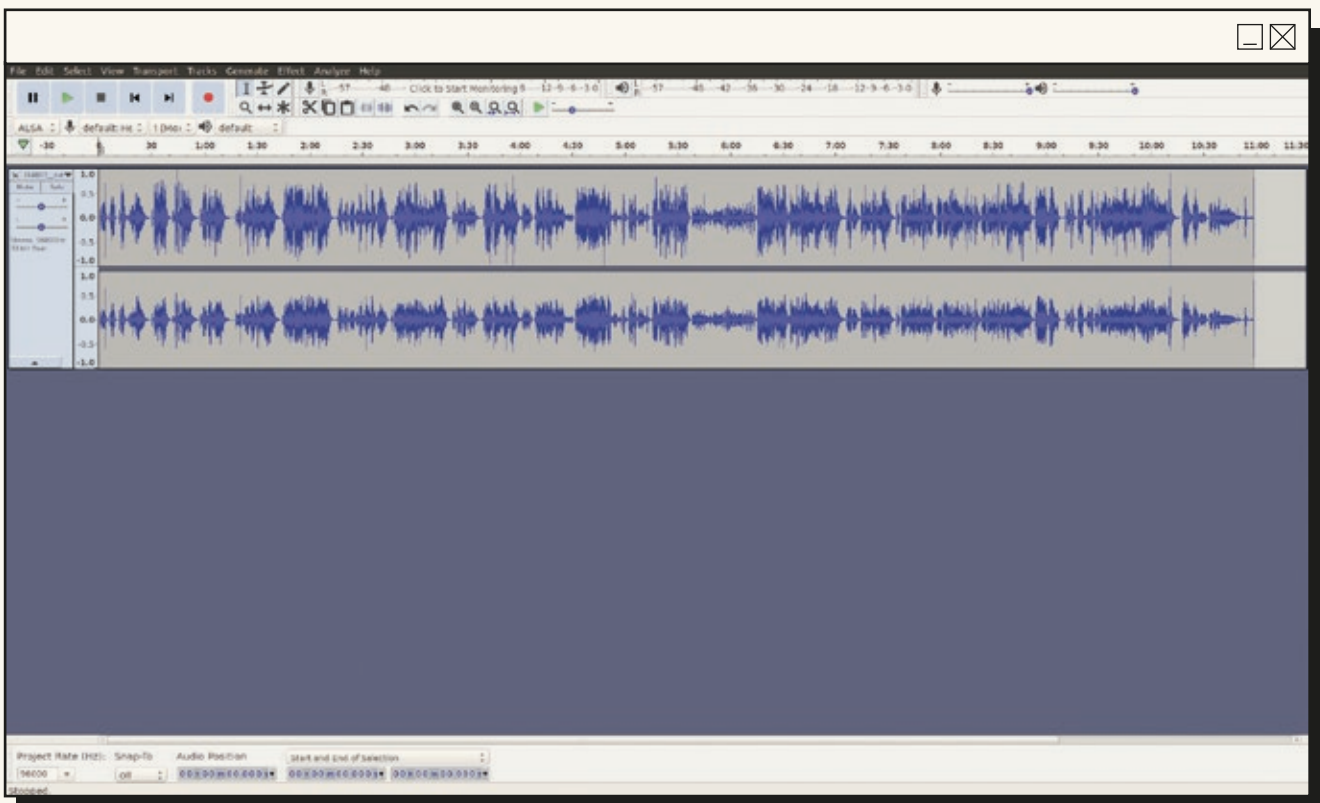
Another way to get .wav or .mp3 files with simple and short sounds is to cut and extract them from long audio files (e.g. songs) using any audio editing software. The Audacity tool (<https://audacity.es>), is recommended, as it is free, open source and cross-platform software, following the steps below:



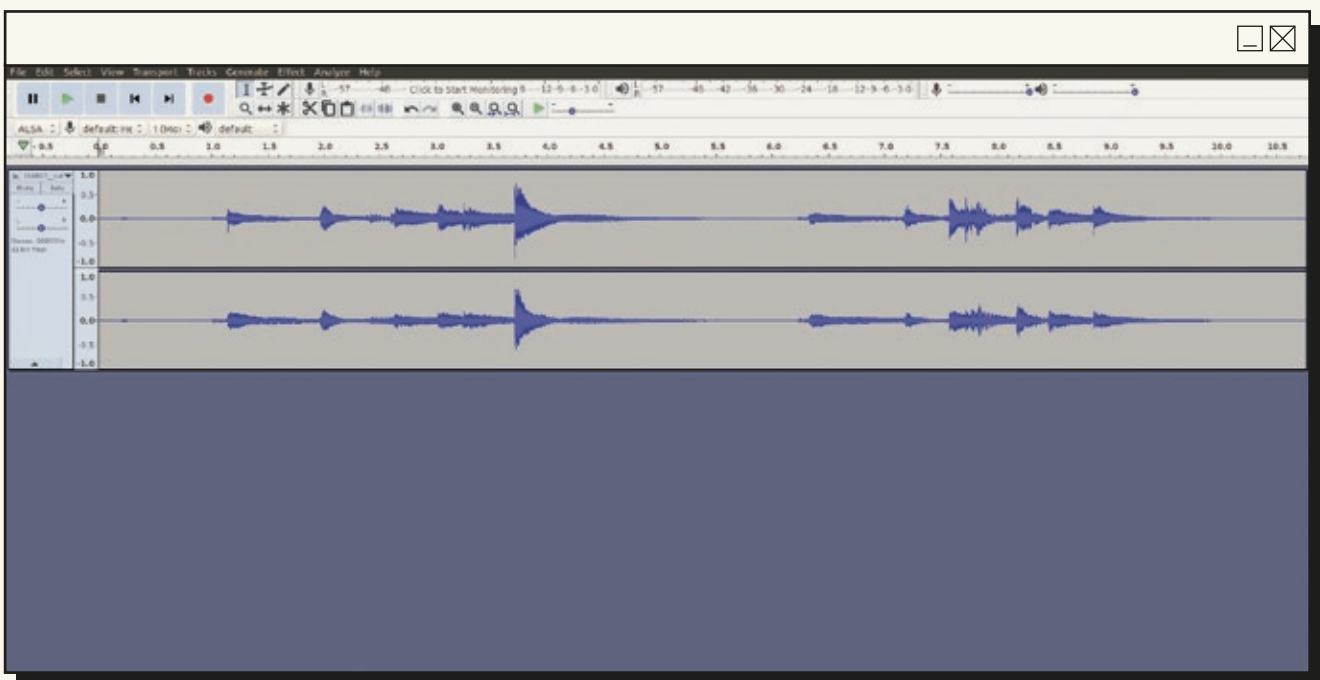
Install Audacity on your computer

(it can be installed and used on Linux, Windows and Mac).

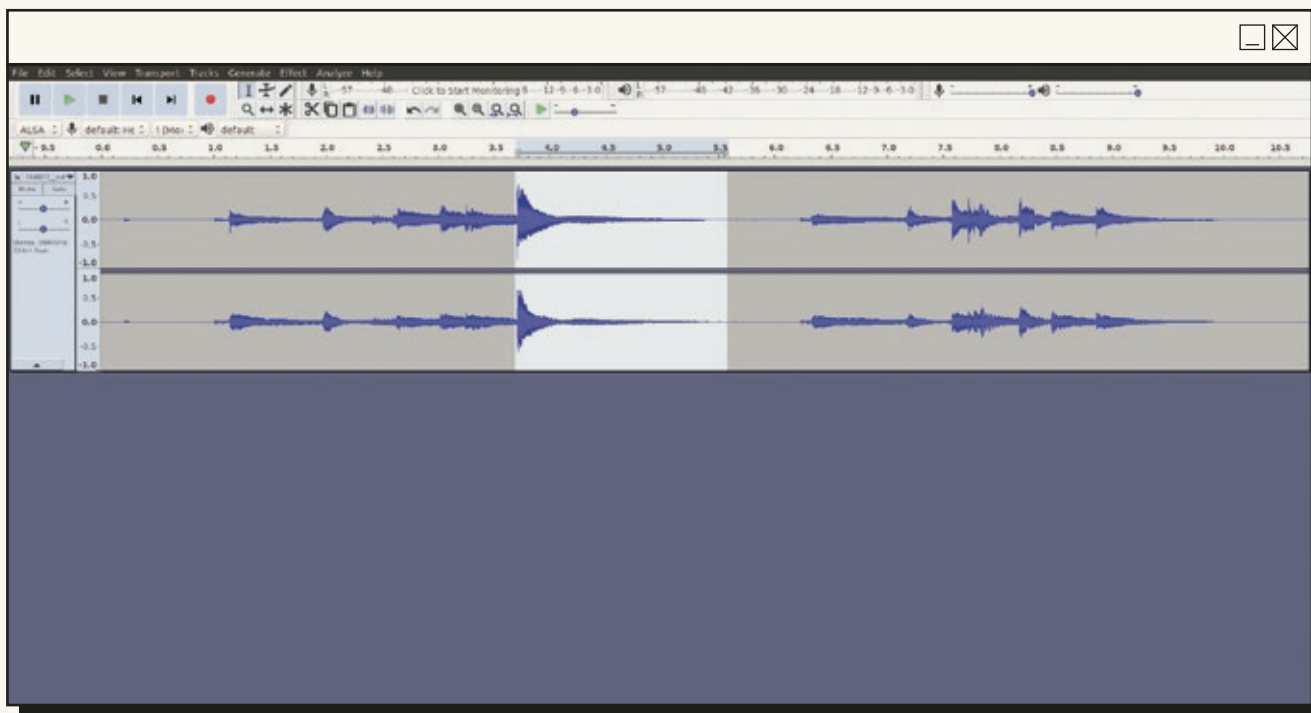
- a) Open Audacity and open the audio file from which you want to cut a simple, short sound.



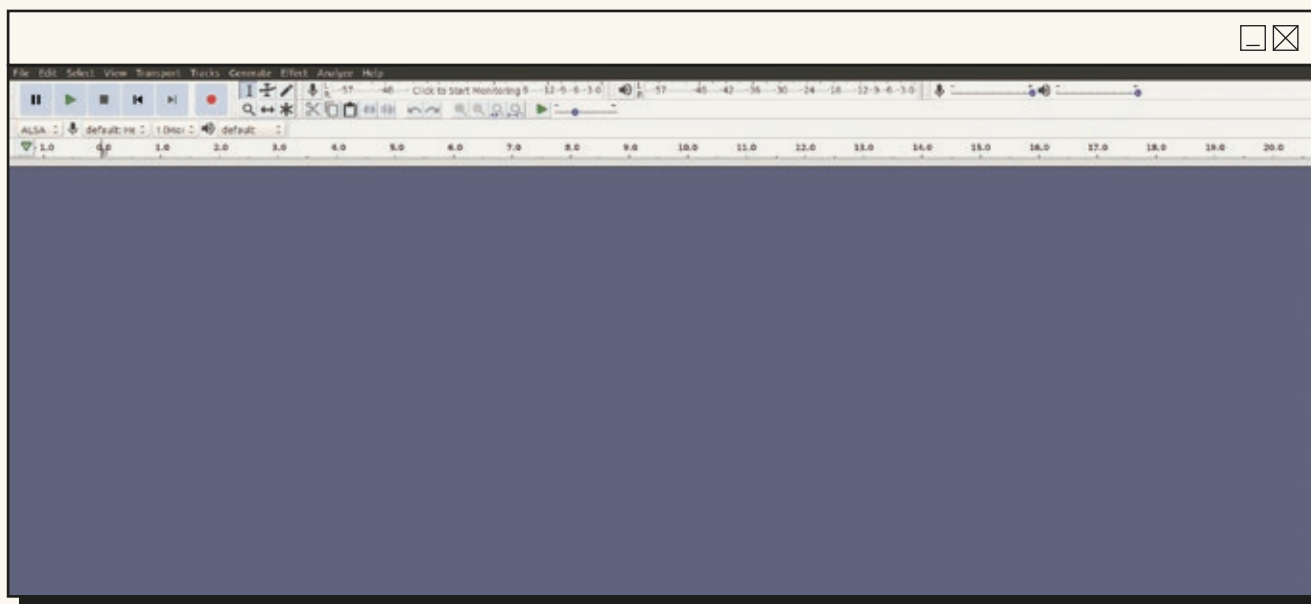
- b) Press “Play” to listen to the audio and thus identify the fragment we want to cut and extract from the audiogram. By previously zooming-in with Ctrl+1 several times, a greater precision is achieved when making the selection. To zoom-out, use Ctrl+3.



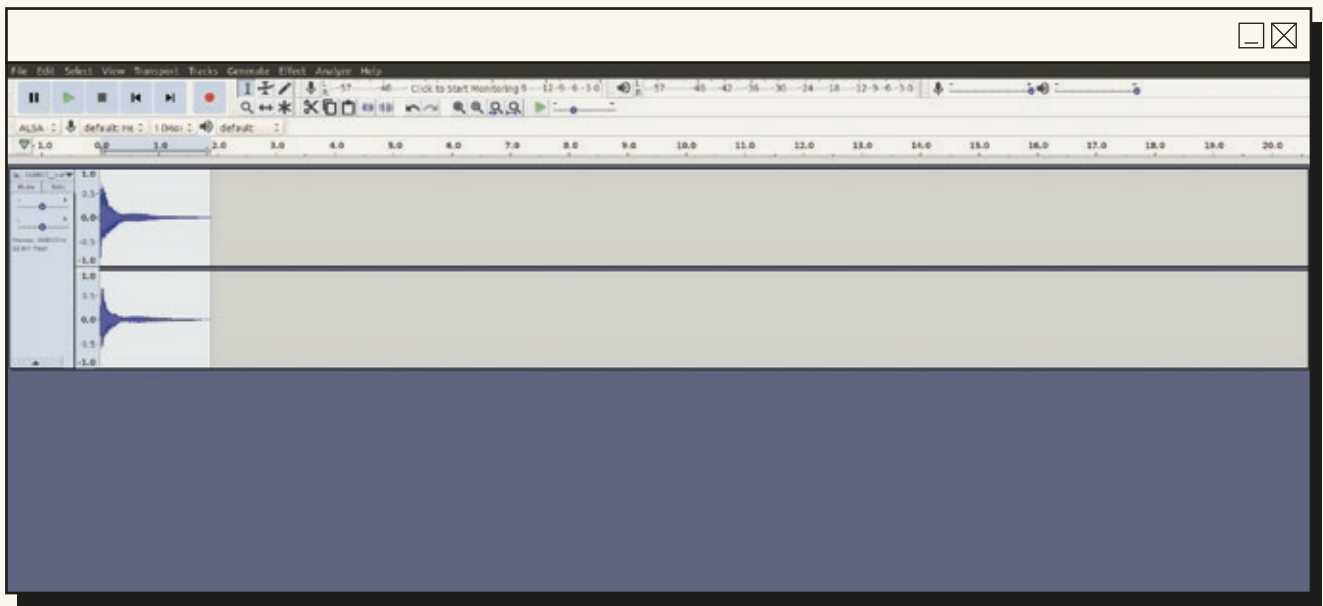
- c) Once the fragment to be trimmed and extracted has been identified, select it with the mouse..



- d) Once selected with the mouse, you can use “Play” again to listen only to that fragment and further adjust the selection as many times as necessary to get only the desired fragment to play.
- e) Copy the fragment (“Edit → Copy”).
- f) Open a new audio file (“File → New”).



g) Paste the copied fragment into the new audio (“Edit → Paste”).



- h)** At this point, different Audacity tools can be used to improve the sound obtained (trim ends, fade-in at the beginning, fade-out at the end, amplify or reduce the volume, change the pitch, distort, etc.).
- i)** Export as .wav or .mp3 (“File → Export → Export as WAV” or “File → Export → Export as MP3”, respectively).
- j)** A dialogue box will open where you can select the name of the .wav or .mp3 file you want to create with the trimmed sound and the directory on your computer where you want to save it.
- k)** When you click “Save”, a dialogue box will open where you can choose metadata for the file (artist name, track title, year, style, etc.).
- l)** Finally, clicking “OK” will create the file and save it with the previously selected name and in the previously selected directory.

6.3. Association between sounds and nucleotides

The basis of the sonification of a genetic sequence is the association of each of the four nucleotides of a genetic sequence (A, T, C and G) to one of the four sounds selected from one or more different instruments, percussion instruments and/or voices. It is this association that makes it possible to generate a sound sequence in which the order of the sounds is determined by the order of the nucleotides in the genetic sequence.

The way sonificADN performs this association is by placing the four files with the four sounds of an instrument in a given directory, the four percussion sounds in second directory and the four voice sounds in third directory, and renaming those files to A4.wav, C4.wav, D4.wav and G4.wav (or A4.mp3, C4.mp3, D4.mp3 and G4.mp3). The reason for having to use these names is that the Python ‘musicpy’ library requires these files to have note names in Anglo-Saxon notation (A, B, C, D, E, F and G instead of Do, Re, Mi, Fa, Sol, La, Si and Do, respectively) and to include the octave value of the note (even if the sound contained in these files does not correspond to that note or octave).



The programme associates:

- the adenine nucleotide (A) with the sounds from files named A4.wav (or A4.mp3)
- the cytosine nucleotide (C) with the sounds from files named C4.wav (or C4.mp3)
- the guanine nucleotide (G) with the sounds from files named G4.wav (or G4.mp3)
- and thymine nucleotide (T) with the sounds from files named D4.wav (or D4.mp3)

Although in three of the four associations the letter designating the nucleotide coincides with the letter designating the supposed (but not real) note in the corresponding file, the choice of these associations is completely arbitrary.

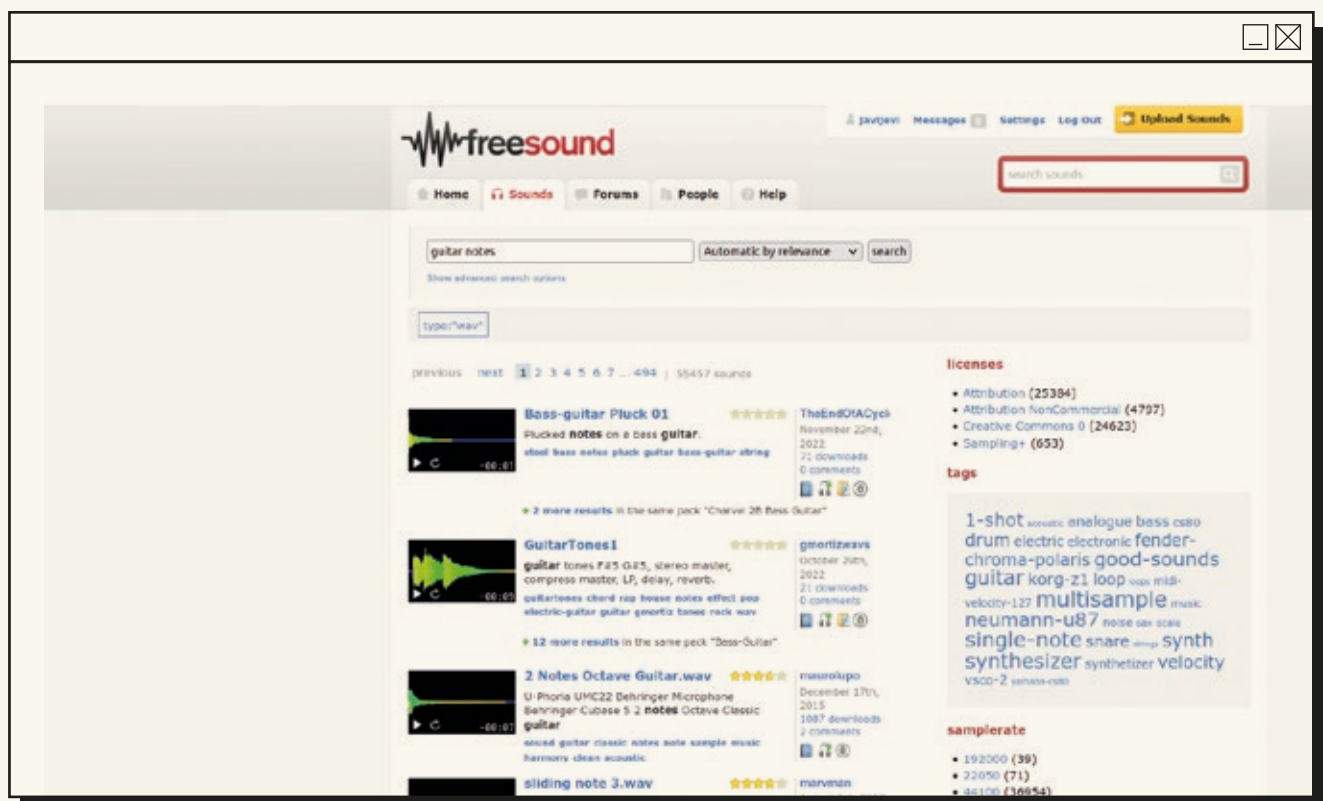
Thus, the files with percussion sounds possibly downloaded in section “3.2. Obtaining sounds for use in sonification” must be renamed to A4.wav, C4.wav, D4.wav and G4.wav (or A4.mp3, C4.mp3, D4.mp3 and G4.mp3), regardless of the note and octave they contain, and placed in the same directory on the computer. This directory should be different from the one used for instrument sounds and from the one used for voice sounds.

Similarly, the files with instrument sounds possibly downloaded in section “3.2. Obtaining sounds for use in sonification” must be renamed to A4.wav, C4.wav, D4.wav and G4.wav (or A4.mp3, C4.mp3, D4.mp3 and G4.mp3), regardless of the note and octave they contain, and placed in the same directory on the computer. Again, this directory should be different from the one used for percussion sounds and from the one used for voice sounds.

Finally, the files with voice sounds possibly downloaded in section “3.2. Obtaining sounds for use in sonification” must be renamed to A4.wav, C4.wav, D4.wav and G4.wav (or A4.mp3, C4.mp3, D4.mp3 and G4.mp3), regardless of the note and octave they contain, and placed in the same directory on the computer. Again, this directory should be different from the one used for percussion sounds and from the one used for instrument sounds.

The paths to these directories with the sounds to be used in the sonification should be indicated in the configuration file 'config.txt', where other parameters of the program are specified, such as the name of the .mp3 file to be generated, the speed of the rhythm of the sound sequence to be generated, the path of the file with the genetic sequence to be sonified, the percentage of sounds to be muted to avoid an excessively monotonous sound sequence and the desired volume for the percussion, for the instrument and for the voices (see section "5.1. Configuration file").

In the configuration file that comes with the .zip downloaded in the section "1. Software", the paths to the directories with the sounds provided as an example are indicated.

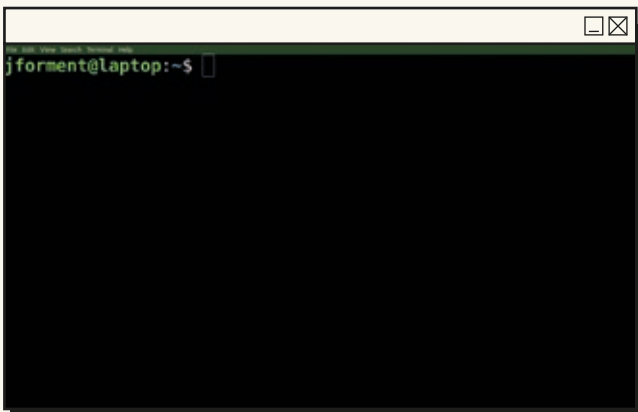


6.4. How to run the programme

Once the above steps have been followed, the way to run the 'sonificADN' program is to use the Linux terminal (also known as the "command line", "shell" or "bash"). Although using the Linux terminal may seem complex at first, it really is not that complex. It is just an environment that you may not be used to. For a successful result, follow all the indicated steps, typing exactly the indicated commands without omitting or adding anything, and remembering that upper and lower case are not equal, and that all the spaces and forward slash symbols (/) are strictly necessary.

To perform the sonification of the example gene sequence using the sample sounds provided in the downloaded .zip file, the steps are as follows:

- a) Open a Linux terminal (also called a "command line"). There are several ways to do this in Ubuntu (most of them also work in other Linux distributions), such as pressing ALT+F2, typing 'gnome-terminal' in the box that opens and hitting 'Enter'.



- b) Use the 'cd' command to move in the terminal to the directory 'sonificADN' that has been created by unzipping the .zip file with the program. For example, if this directory is in the path /home/user/sonificADN (where 'user' is the user with which we are logged into Linux), the command would be to type the following in the terminal and hit 'Enter':

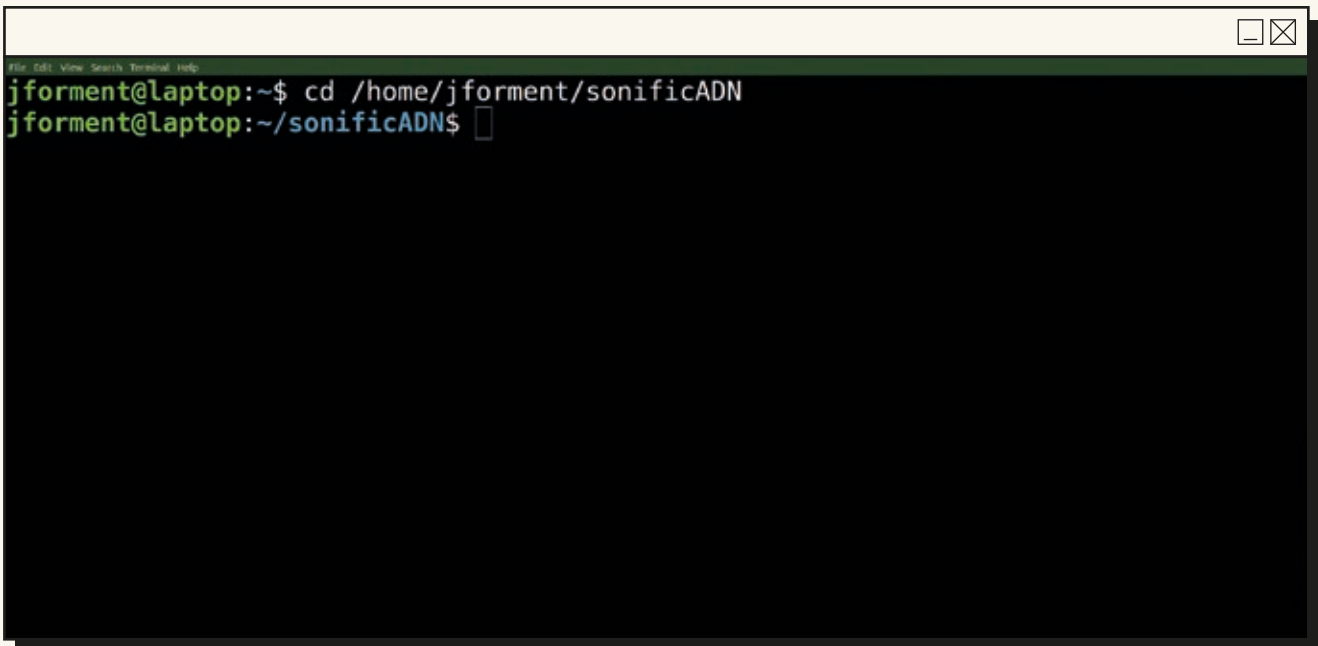
cd /home/jforment/sonificADN

If you cannot remember where you unzipped the .zip file, you can locate the 'sonificADN' directory with the 'locate' command by typing the following in the terminal and hitting 'Enter':

locate sonificADN

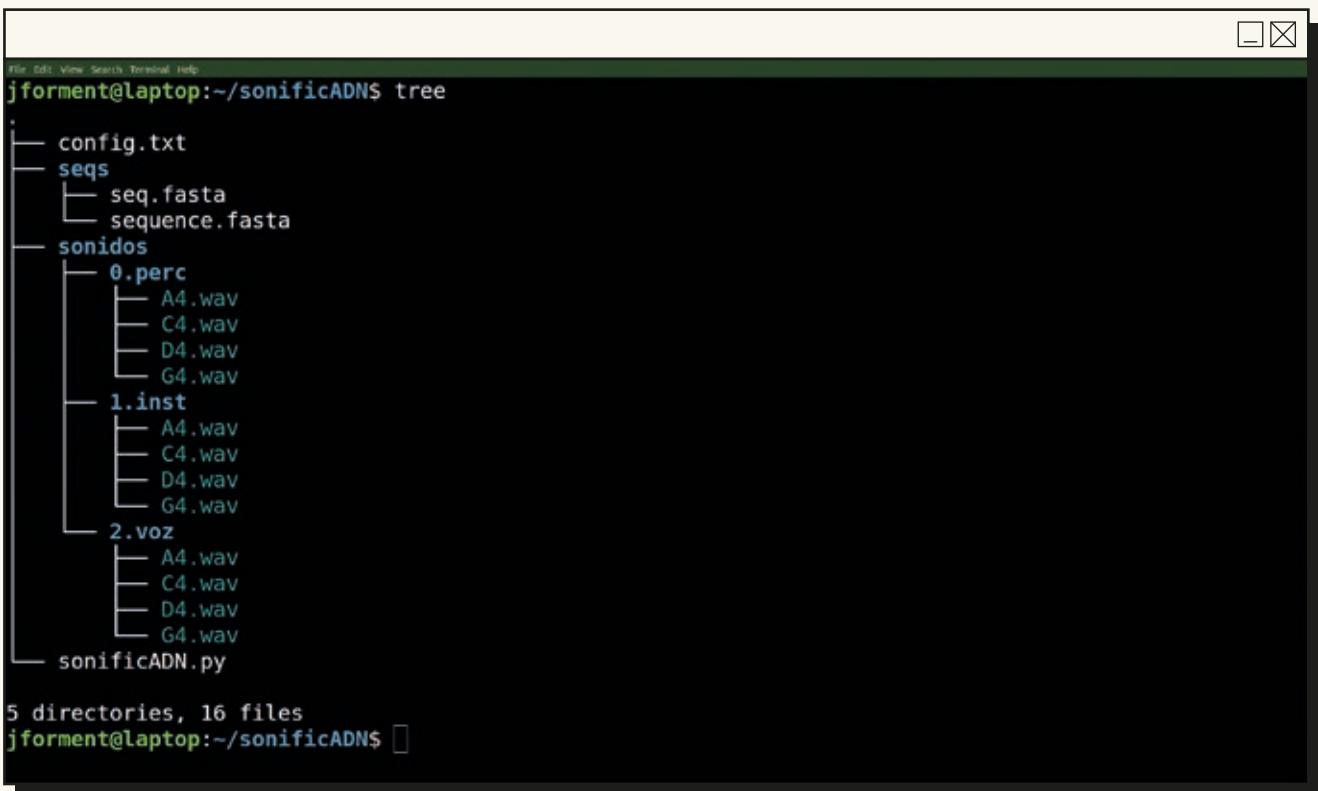
The first path that appears is the path to use with the 'cd' command (instead of '/home/user/classes/sonificADN' in the example above).

- c) Once correctly located in the 'sonificADN' directory (the path ending in 'sonificADN' will appear partially in the terminal, just to the left of the cursor).



```
File Edit View Search Terminal Help
jforment@laptop:~$ cd /home/jforment/sonificADN
jforment@laptop:~/sonificADN$
```

- d) We can check that we have all the files necessary to run the program with the 'tree' command, which shows us the structure of files and subdirectories.



```
File Edit View Search Terminal Help
jforment@laptop:~/sonificADN$ tree
.
├── config.txt
├── seqs
│   ├── seq.fasta
│   └── sequence.fasta
├── sonidos
│   ├── 0.perc
│   │   ├── A4.wav
│   │   ├── C4.wav
│   │   ├── D4.wav
│   │   └── G4.wav
│   ├── 1.inst
│   │   ├── A4.wav
│   │   ├── C4.wav
│   │   ├── D4.wav
│   │   └── G4.wav
│   └── 2.voz
│       ├── A4.wav
│       ├── C4.wav
│       ├── D4.wav
│       └── G4.wav
└── sonificADN.py

5 directories, 16 files
jforment@laptop:~/sonificADN$
```

- e) It is possible that the ‘tree’ command is not installed in our Linux, in which case we can check the structure of files and subdirectories with ‘ls -R’ (although it shows it in a less graphical and intuitive way).



```
File Edit View Search Terminal Help
jforment@laptop:~/sonificADN$ ls -R
.:
config.txt  seqs  sonidos  sonificADN.py

./seqs:
seq.fasta  sequence.fasta

./sonidos:
0.perc  1.inst  2.voz

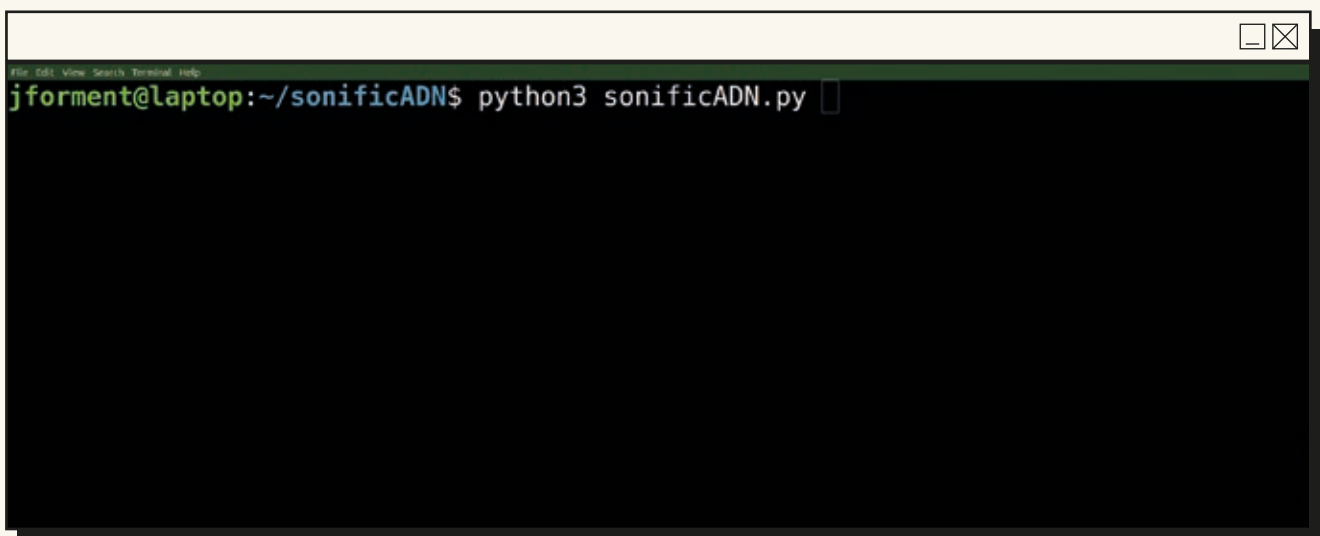
./sonidos/0.perc:
A4.wav  C4.wav  D4.wav  G4.wav

./sonidos/1.inst:
A4.wav  C4.wav  D4.wav  G4.wav

./sonidos/2.voz:
A4.wav  C4.wav  D4.wav  G4.wav
jforment@laptop:~/sonificADN$
```

- f) Once you have verified that you are in the correct directory, you can run the program by typing the following command and hitting ‘Enter’:

python3 sonificADN.py



```
File Edit View Search Terminal Help
jforment@laptop:~/sonificADN$ python3 sonificADN.py
```

- g) The following messages will appear successively on the terminal (the speed will depend on the length of the sequence used and the power of the computer used):

rendering track 1/3 channel 1

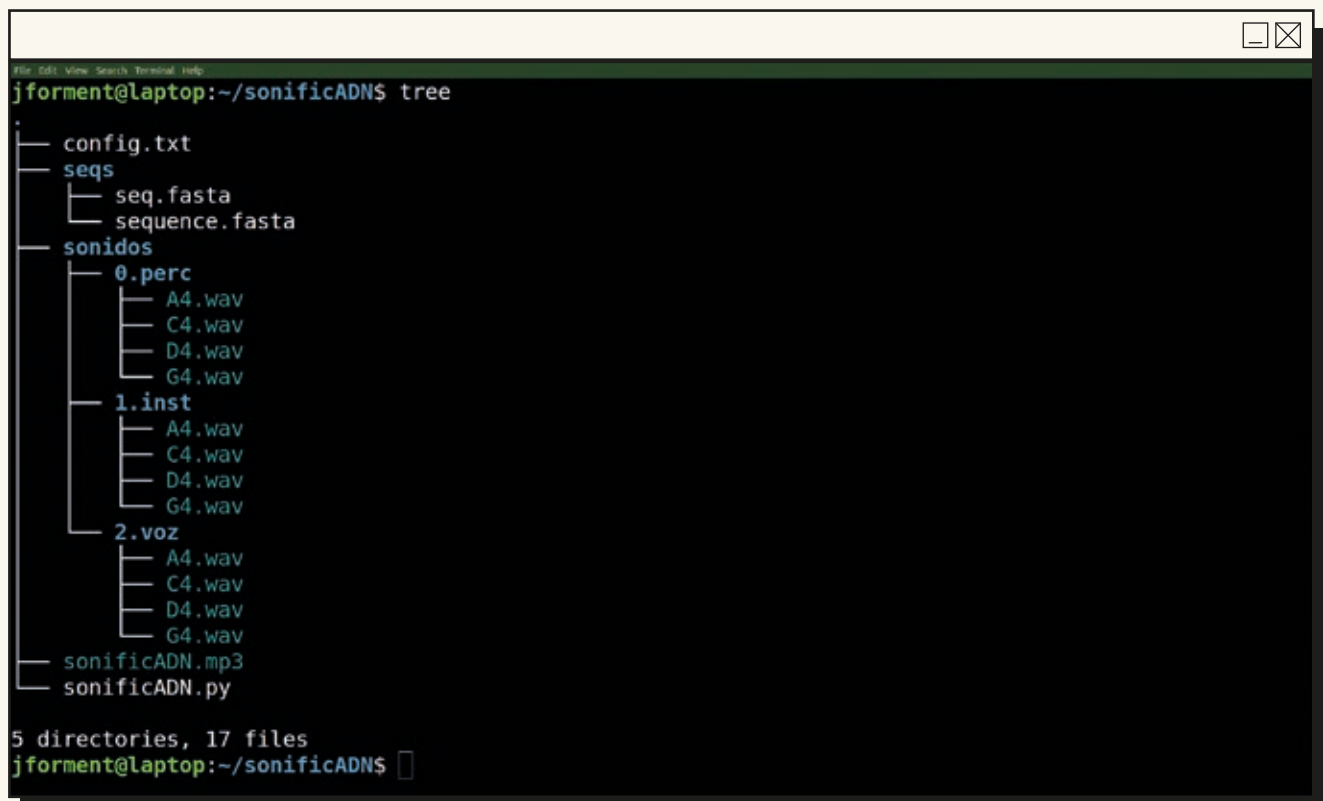
rendering track 2/3 channel 2

rendering track 3/3 channel 3

rendering finished

export finished

- h) After that, the cursor will reappear (with the partial path ending in 'sonificADN' just to the left of it), indicating that the programme has finished. In the directory we are in ('sonificADN') an .mp3 file containing the sonification will have appeared. This file will have the name 'sonificADN.mp3' (or the one we have indicated in the configuration file 'config.txt' if we have changed it). We can check it again with the command 'tree' or 'ls -R', which indicate the names of the files and directories contained in the directory we are in:



```
File Edit View Search Terminal Help
jforment@laptop:~/sonificADN$ tree
.
├── config.txt
├── seqs
│   ├── seq.fasta
│   └── sequence.fasta
├── sonidos
│   ├── 0.perc
│   │   ├── A4.wav
│   │   ├── C4.wav
│   │   ├── D4.wav
│   │   └── G4.wav
│   ├── 1.inst
│   │   ├── A4.wav
│   │   ├── C4.wav
│   │   ├── D4.wav
│   │   └── G4.wav
│   └── 2.voz
│       ├── A4.wav
│       ├── C4.wav
│       ├── D4.wav
│       └── G4.wav
├── sonificADN.mp3
└── sonificADN.py

5 directories, 17 files
jforment@laptop:~/sonificADN$
```



```
File Edit View Search Terminal Help
jforment@laptop:~/sonificADN$ ls -R
.:
config.txt  seqs  sonidos  sonificADN.mp3  sonificADN.py

./seqs:
seq.fasta  sequence.fasta

./sonidos:
0.perc  1.inst  2.voz

./sonidos/0.perc:
A4.wav  C4.wav  D4.wav  G4.wav

./sonidos/1.inst:
A4.wav  C4.wav  D4.wav  G4.wav

./sonidos/2.voz:
A4.wav  C4.wav  D4.wav  G4.wav
jforment@laptop:~/sonificADN$
```

- i) We can listen to this .mp3 file with the sonification of the gene sequence on any device that plays .mp3 files.

6.5. Configuration file

If we want to use another genetic sequence different from the one provided as an example, and which we have previously downloaded (as indicated in previous sections), and/or if we want to use other sounds different from those provided as an example, and which we have previously downloaded (as indicated in previous sections), we must edit the configuration file 'config.txt' to indicate to the programme where this sequence and/or these sounds are located. To do this, follow the steps below:

- Open the configuration file 'config.txt' with a text editor. On Ubuntu you can use the graphical text editor 'gedit'.
- Change the lines starting with 'seqfasta' and/or 'percdir', 'instdir' and/or 'vozdir' so that instead of 'seqs/sequence.fasta', 'sounds/0.perc', 'sounds/1.inst' and/or 'sounds/2.voz' they contain the path to the file with the genetic sequence we want to use, and/or the path to the directories where the files with the sounds we want to use are located, respectively.

In this configuration file we can also change other characteristics of the desired sonification:

- the name of the resulting .mp3 file: on the line beginning with 'mp3file'.
- the beat rate (in bpm's, "beats per minute") of the resulting sound piece: on the line beginning with 'bpm'.
- the percentage of notes, percussion hits and voices that we want to silence in the final sound piece to make it less monotonous and more musically aesthetic: on the line starting with 'silences'.
- the volume of instrument notes, percussion hits and vocals: on the lines beginning with 'volumeperc', 'volumeinst' and 'volumevoice', respectively.

Experimentation with such changes is recommended. For example:

- see what results from muting the instrument, or the voice, or the percussion (or two of them) completely, setting their volume to 0
- see what happens by changing the tempo of the song ('bpm' parameter)
- see what results from changing the percentage of silences
- see what results using gene sequences composed of a single nucleotide repeat (or two nucleotide repeats)
- see what results from using in the 'percdir' parameter a directory that does not contain percussion sounds but another instrument than the one in 'instdir', or another voice than the one in 'vozdir'; or using in 'instdir' and/or 'vozdir' directories with different percussion sounds than the ones in 'percdir'.

sonificADN

*Digital resource for the automatic generation of sound sequences
from genetic sequences*



What does a gene sound like? How is a gene similar to a musical score? Can a gene create a song? This resource offers students and teachers the possibility of establishing and internalising both the concept of a genetic sequence and the concept of a piece of music by establishing a parallelism between a sequence of nucleotides (a gene) and a sequence of sounds (a song) using a computer programme. Given the multidisciplinary nature of the resource, which combines concepts from genetics, music and computer science, we propose its use through collaborations amongst teachers from the areas of Biology, Art and Technology.

This resource is available in English and Spanish
at Centro de recursos Planea

